

Continue



Cracking the coding interview book pdf github

Cracking coding interview. Cracking the coding interview epub github. Cracking the coding interview book near me. Cracking the coding interview pdf github.

Get top software developer jobs with Cracking the Coding Interview's updated 5th edition. This in-depth guide focuses on technical skills needed to ace your interview. Spanning over 500 pages, it includes 150 programming questions and answers, plus expert advice. The book covers essential topics like the interview process, behind-the-scenes insights, special situations for different roles, and behavioral preparation. Additionally, you'll learn how to tackle complex algorithm problems and get actionable tips on "good coding" during an interview. The latest edition boasts an impressive expansion of over 200 pages of fresh content, effectively doubling the book's size from 300 to more than 500 pages. This substantial update encompasses a wide range of revisions, including the introduction of multiple alternative solutions and significant expansions in both the introductory chapters and the Technical Questions sections for each chapter. Additionally, a total of 24 new questions have been incorporated into the guide. Cracking the Coding Interview: Fifth Edition stands as the most comprehensive resource available for acing software development/programming interviews. Authored by Gayle Laakmann McDowell, founder and CEO of CareerCup, this book also includes her expertise in writing "Cracking the PM Interview" and "Cracking the Tech Career." With a distinguished background that includes roles at Microsoft, Apple, and Google as a software engineer, Gayle holds a strong educational foundation from the University of Pennsylvania in Computer Science and an MBA from Wharton. The book provides a set of problems that are typical in coding interviews, with hints and answers included. It's recommended to practice solving the problems on a whiteboard without looking at the answers, as this is similar to how interviewers typically provide hints during an interview. This method emulates the coding interview experience more effectively than simply reading through solutions. The author has mixed feelings about hiring practices in the software industry, stating that many interview questions are overly complex and require memorization rather than developing strong CS and Math skills. Despite this, they found the book to be a useful resource for getting a sense of what to expect during coding interviews. On the other hand, some reviewers felt that the book did not help them crack the interview, while others reported significant improvement after applying techniques learned from the book. The key takeaways include talking out loud during the interview process, using simple examples to aid thinking, decreasing time and space complexity of algorithms, and adopting a logic-based approach to solving problems. Overall, the book is recommended for anyone preparing for coding interviews. Note: The original text has been paraphrased without translating it from its original language. This book is a must-have for software engineers, regardless of their current job status or level of experience. It serves as a valuable resource for understanding how programming interviews are conducted in most software companies. The book is divided into two parts: an overview of the interview process and actual coding exercises with hints and complete answers. The overview section highlights that during these interviews, the majority of time is spent coding on a whiteboard, which can be quite different from coding on a computer. Key takeaways include understanding that the goal of coding in this context is not to demonstrate existing knowledge but rather how one approaches and thinks through unfamiliar problems. It's essential to remember that no one expects immediate or perfect solutions; rather, the focus is on observing how you react to unfamiliar situations. It's also important to note that interviewers grade candidates based on their performance relative to others in the pool, not necessarily on whether they answer every question correctly. The book provides a good foundation for preparation, and its focus can be tailored to specific companies or types of interviews being faced. A user found this book particularly useful, especially when preparing for technical interviews. They noted that brushing up on the Java section before an interview led to them being asked about a relevant topic, which they were prepared for thanks to the refresher. This experience earned the book a 4-star rating from the user. November 9, 2013 & November 25, 2017 & August 30, 2018 The book provides insights into the coding interview process, highlighting the importance of approaching problems rather than focusing on specific solutions. It offers valuable lessons and explanations to help readers improve their problem-solving skills. Many people study Computer Science for the wrong reasons, leading to issues like increased homogeneity in the field. The author doesn't blame Gayle or companies that use classic problems as a screening tool. The book "Cracking the Coding Interview" is a classic and still widely recommended, despite being old-fashioned. It's great for practicing old topics, but readers may need more detailed books on specific subjects. Some issues with the book include missing comments when code is lengthy, making it difficult to understand. This can be frustrating when trying to decipher complex programs. Despite this, the book remains a valuable resource for coding interview preparation and practicing old topics. I would say this book is not bad overall, but it has its downsides. The OOP section contains some subpar solutions, like the car park example. On the other hand, the book itself is great, with the exception maybe being the Test chapter, which might be better skipped if you're from a testing background. The DP programming questions and their accompanying solutions also leave much to be desired. I think one of the biggest issues with this book is its tendency to overanalyze solution steps. In some cases, it made the solution harder than it needed to be. I found that writing my own code and figuring out how it works was more beneficial for me than following the provided solutions. Sometimes, my own written code even looked cleaner and easier to understand. My suggestion would be not to read this book cover-to-cover; it's just too much information all at once. However, if you're looking for a resource that covers everything from coding to behavioral questions, then maybe this is the right book for you. Just remember, practice makes perfect, so make sure to put what you learn into action. Some people might find the solutions provided in this book very helpful, and that's okay. But personally, I found them to be sometimes confusing, especially when compared to my own written code. Perhaps a more balanced approach would be beneficial for many readers. This book offers valuable insights into tackling problems and preparing for coding interviews, covering key concepts such as Big O time complexity. However, some users find LeetCode a more effective platform for practice due to its wider variety of problems. A recommended approach is to start with the introductory chapters and then transition to practicing on LeetCode. For those looking for an essential resource in coding interview preparation, this book is highly recommended. It not only provides excellent questions but also guides readers through breaking down complex problems into manageable steps. The author's explanations are clear, making it easy to tackle previously unfamiliar problems. The targeted audience includes software engineers and data scientists, with the initial chapters offering general information applicable to both fields. These sections cover interview processes, company differences, behavioral questions, and more. However, the majority of the book focuses on coding topics that mainly interest software engineers. This is a great resource for preparing coding interviews, especially in large corporations like Google or Amazon. It provides concrete steps for preparation and devotes about 70% of its content to detailed problem solutions. A significant amount of time may be spent solving these problems rather than reading the book itself. Some reviewers find this book realistic and easy to read, particularly for those majoring in Computer Science or preparing for job interviews. The book includes good sample questions for whiteboarding interviews but lacks examples of non-whiteboarding interviews typical of smaller or medium-sized companies. Despite being a great coding interview prep book, CtCi is slightly overrated according to some users. It's recommended to read it alongside other books due to its lack of explanation in certain parts. Overall, this book is suitable for those who have not previously read any coding interview prep material. Coding interviews? No problem! This book by Gayle Laakmann McDowell gives you a straightforward guide on what matters and what doesn't. It's not just for recruiters, but also for software engineers like me who know how it feels to be thrown curveballs during an interview. I've been in both shoes - as a candidate and as an interviewer. That's why I created this book, drawing from my experience coaching and interviewing hundreds of software engineers. Want to ace those coding interviews? This book teaches you how to: * Uncover hidden details and hints in questions * Break down complex problems into manageable chunks * Get unstuck when you're stuck * Learn core computer science concepts * Practice on 189 real interview questions with solutions These questions aren't from textbooks; they're based on what top companies like Google and Facebook actually ask. You'll learn how to derive each solution, just like in a real interview. Plus, get hints on solving each question and five proven strategies for tackling algorithm problems. The book also covers essential topics like big O time, data structures, and core algorithms. And, it gives you a behind-the-scenes look at how top companies hire developers. Oh, and don't forget about the soft side of interviews - behavioral questions! The book even provides insights for interviewers and companies on what makes a good interview question and hiring process. **Gayle Laakmann McDowell is the Founder/CEO of CareerCup.com and author of two books: Cracking the Coding Interview, 6th Edition.** A top-selling author of interview books, including "The Google Resume", this expert has held positions at industry giants Google, Microsoft, and Apple. She also played a key role on Google's hiring committee. With degrees in Computer Science from the University of Pennsylvania and an MBA from Wharton School, she brings a wealth of knowledge to her work.